matching measures generally increases with proximity to the subject element of the input sentence. Those skilled in the art will be able to find suitable values for the weights by trial and error.

5   The second innermost loop of instructions then supplements the sequence similarity measure by taking into account the extent to which the boundaries (if any) already predicted for the input sentence match the boundaries present in the reference sequence. Only the part of the search sequence before the subject element is considered since no boundaries have yet been predicted for the subject element or
10  the elements which follow it. Pseudo-code for the second innermost loop of instructions is given below:

**FOR current_position_in_srch_seq ( = q) = srch_seq_start to m-1**
         **s.s.m = s.s.m + weight(q) * bdymatch(srch_element_q, corres_ref_element)**
15  **NEXT**

The boundary matching measure between two elements (expressed in the form bdymatch(element x,element y) in the above pseudo-code) is set to two if both the input sentence and the reference sentence have a boundary of the same type after
20  the qth element, one if they have boundaries of different types, zero if neither has a boundary, minus one if one has a minor boundary and the other has none, and minus two if one has a strong boundary and the other has none. A weighted addition of the boundary matching measures is applied, those inter-element boundaries close to the current element being given a higher weight. The weights are chosen so as to
25  penalise heavily sentences whose boundaries do not match.

It will be realised that the carrying out of the first and second innermost loop of instructions results in the generation of a sequence similarity measure for the subject element of the input sentence and the reference element of the corpus 52. If the
30  sequence similarity measure is the highest yet found for the subject element of the input sentence, then the best match value is updated to equal that measure (step 116) and the number of the associated element is recorded (step 118).

Once the final element has been compared, the computer ascertains whether the core element in the best matching sequence has a boundary after it. If it does, a boundary of a similar type is placed into the input sentence at that position (step 122).

5

Thereafter a check is made to see whether the current element is now the final element (step 124). If it is, then the prosodic structure prediction process 50 ends (step 126). The boundaries which are placed in the input sentence by the above prosodic boundary prediction process (Figure 5) constitute the phrase boundary data (Figure 2A : 54). The remainder of the text-to-speech conversion process has already been described above with reference to Figure 2B.

10

In a preferred embodiment of the present invention, boundaries are predicted on the basis of the ten best matching sequences in the prosodic structure corpus. If the majority of those ten sequences feature a boundary after the current element then a boundary is placed after the corresponding element in the input sentence.

15

In the above-described embodiment pattern matching was carried out which compared an input sentence with sequences in the corpus that included sequences bridging consecutive sentences. Alternative embodiments can be envisaged, where only reference sequences which lie entirely within a sentence are considered. A further constraint can be placed on the pattern matching by only considering reference sequences that have an identical position in the reference sentence to the position of the search sequence in the input sentence. Other search algorithms will occur to those skilled in the art.

20

25

The description of the above embodiments describes a text-to-speech program being loaded into the computer from a CD-ROM. It is to be understood that the program could also be loaded into the computer via a computer network such as the Internet.

30

22

CLAIMS

1.      A method of converting text to speech comprising the steps of:

5          receiving an input word sequence in the form of text;

comparing said input word sequence with each one of a plurality of reference word sequences provided with prosodic boundary information;

identifying one or more reference word sequences which most closely match said input word sequence; and

10          predicting prosodic boundaries for a synthesised spoken version of the input text on the basis of the prosodic boundary information included with said one or more most closely matching reference word sequences.

2.      A method according to claim 1 further comprising the step of:

15          identifying clusters of words in the input text which are unlikely to include prosodic phrase boundaries;

wherein:

said plurality of reference sentences are further provided with information identifying such clusters of words therein; and

20          said comparison step comprises a plurality of per-cluster comparisons.

3.      A method according to claim 2 wherein said per-cluster comparison comprises quantifying the degree of similarity between the syntactic characteristics of the clusters.

25

4.      A method according to claim 2 wherein said per-cluster comparison comprises quantifying the degree of similarity between the syntactic characteristics of the words within the clusters.

30 5.      A method according to claim 2 wherein said per-cluster comparison comprises measuring the difference in the number of words in the clusters being compared.